



## СПбГЭТУ «ЛЭТИ» первый электротехнический



Р.Р. Фаткиева

# Интернет вещей

Передача данных с датчика подключенного к Arduino

СПбГЭТУ «ЛЭТИ», 2021 г.





### 2. ПРОТОКОЛЫ ПЕРЕДАЧИ ДАННЫХ. ПЕРЕДАЧА ДАННЫХ С ДАТЧИКА ПОДКЛЮЧЕННОГО К ARDUINO

Цель работы: изучение теоретического материала о протоколах передачи данных, формирование практических навыков передачи данных с использованием технологии Bluetooth.

#### 2.1. Задание

Осуществить передачу данных с датчика, подключенного к Arduino на интерфейс SPI платы Raspberry Pi.

#### 2.2. Порядок выполнения работы

Шаг 1. Осуществляем подключение модуля Bluetooth HC 06 (Рис. 1) к Arduino Nano согласно схеме, представленной на Рис. 2, используя макетную плату. Модуль Bluetooth HC 06 имеет основные характеристики:

- Питание 3,3В 6 В;
- Максимальное входное напряжение 5 B;
- Максимальный ток 45 мА;
- Скорость передачи данных 1200-1382400 бод;
- Рабочие частоты 2,40 ГГц -2,48ГГц;
- Поддержка спецификации bluetooth версии 2.1;
- Малое потребление энергии;
- Высокий уровень защиты данных;
- Дальность связи 30 м;



#### Рис. 1 [1]

1.1. Разрабатываем «прошивку» Bluetooth HC 06 с использованием программной среды Arduino IDE и скетча (Приложение 1)

1.2 Реализуем подключение Bluetooth модуль к микроконтроллеру Arduino согласно табл. 1 или схеме на Рис. 2

Табл.1





Arduino	Bluetooth
Pin 1 (TX)	RXD
Pin 0 (RX)	TXD
GND	GND
5V	VCC
C-D5 Wetooth RXD State	DIR DIR REF AD AL AP AS AL A7 SV RST GND VIN

Рис.2

1.3 При использовании макетной платы подключение выглядит как показано на рис. 3.



Рис.3

Шаг 2. Для взаимодействия модуля Bluetooth и передачи данных с микроконтроллера Arduino необходимо сформировать скрипт, с использованием программной среды Arduino IDE, который представлен в Приложении 2. Код позволяет считать состояние шины с входа SPI и





транслировать все через UART на Bluetooth. Так же способен выступать в роли самостоятельного передатчика по Bluetooth значений, которые заданы скриптом.

Шаг 3. Осуществляем подключение Arduino Nano к Raspberry Pi посредством последовательного интерфейса SPI. В отличие от некоторых схем последовательной шины, где линия передачи (TX) от одного устройства подключена к линии приема (RX) на другом, в SPI четыре линии на одном устройстве подключены непосредственно к их аналогу на другом (Puc .4 [2]).



3.1 Подключаем Arduino Nano к Raspberry Pi согласно схеме: Ha Arduino Nano:

- SS находится на физическом выводе 16,
- MOSI на выводе 17,
- MISO на выводе 18
- SCLK на выводе 19.

Указанные физические контакты соответствуют цифровым пинам Arduino с 10 по 13.

Для Raspberry Pi это вывод 19, вывод 21 MISO и вывод 23 SCLK Рис 5 [2]







Рис.5 [2]

3.2 Ha Raspberry Pi включаем поддержку GPIO и SPI:

Для того, чтобы активировать интерфейс SPI платы Raspberry Pi можно воспользоваться утилитой *raspi-config*. Ввести команду:

sudo raspi-config

Далее следуя по пути Interfacing options - SPI и установить необходимы интерфейс Р4. После изменения конфигурации необходимо перезапустить плату:

sudo reboot

Для проверки вводим команду:

ls /dev

Наличие в списке интерфейсов spidev0.0 и spidev0.1 сигнализирует нам о том, что интерфейс SPI успешно активирован.

Шаг 4. Запускаем скрипт передачи данных с датчика на Arduino.

Шаг 5. Разрабатываем и проводим тестирование с использованием Android приложения (Получение данных происходит посредством установки Bluetooth соединения). В качестве примера можно использовать [3].





#### КОНТРОЛЬНЫЕ ВОПРОСЫ

- 1. В чем преимущество передачи данных по SPI ? Перечислите виды цифровых сигналов.
- 2. Как происходит синхронизация сигнала? Для чего она необходима?
- 3. Уважите режимы работы SPI?
- 4. Перечислите преимущества и недостатки SPI?
- 5. Какие порты использует передача по SPI?





Приложение 1. Программный код для взаимодействия модуля Bluetooth [1]

```
char incomingbyte;
void setup() {
 pinMode(ledPin, OUTPUT);
 Serial.begin(9600);
}
void loop() {
 if (Serial.available() > 0) {
  incomingbyte = Serial.read();
    switch(incomingbyte) {
    case '1':
      digitalWrite(ledPin, HIGH);
      break;
   case '0': // если приходит "0"
      digitalWrite(ledPin, LOW);
      break;
   }
}
}
```

const int ledPin = 13;





Приложение 2. Программный код для взаимодействия модуля Arduino

Nano [2]

Raspberry Pi [2]

```
unsigned char hello[] = {'H', 'e', 'l', 'l', 'o', ' ',
                  'R','a','s','p','i','\n'};
byte marker = 0;
void (void)
{
  pinMode(MISO, OUTPUT);
 SPCR |= _BV(SPE);
}
void (void)
{
 if((SPSR & (1 << SPIF)) != 0)
 Ł
  SPDR = hello[marker];
  marker++;
  if(marker > sizeof(hello))
  {
    marker = 0;
  }
 }
}
           Приложение 2. Программный код для взаимодействия модуля
using namespace std;
int fd;
unsigned char hello[] = {'H', 'e', 'l', 'l', 'o', ' ',
                   'A','r','d','u','i','n','o'};
unsigned char;
int spiTxRx(unsigned char txDat);
int main (void)
{
  fd = open("/dev/spidev0.0", O_RDWR);
  unsigned int speed = 1000000;
  ioctl (fd, SPI_IOC_WR_MAX_SPEED_HZ, & speed);
  while (1)
  {
    for (int = 0; i < sizeof(hello); i++)</pre>
    £
      result = spiTxRx(hello [i]);
      cout << result;</pre>
      usleep (10);
    }
 }
}
```





```
int spiTxRx(unsigned char txDat)
{
    unsigned char rxDat;
    struct spi_ioc_transfer spi;
    memset (&spi, 0, sizeof (spi));
    spi.tx_buf = (unsigned long)&txDat;
    spi.rx_buf = (unsigned long)&trxDat;
    spi.len = 1;
    ioctl (fd, SPI_IOC_MESSAGE(1), &spi);
    return rxDat;
}
```





#### СПИСОК ЛИТЕРАТУРЫ

- 1. КакподключитькArduinoмодульBluetooth.<a href="https://soltau.ru/index.php/arduino/item/400-kak-podklyuchit-k-arduino-modul-bluetooth">https://soltau.ru/index.php/arduino/item/400-kak-podklyuchit-k-arduino-modul-bluetooth</a>
- 2. СвязьRaspberryPicArduinoSPI.<a href="http://robotics.hobbizine.com/raspiduino.html">http://robotics.hobbizine.com/raspiduino.html</a>
- 3. Bluetooth модуль HC-06 подключение к Arduino. Управление устройствами с телефона <u>https://lesson.iarduino.ru/page/bluetooth-</u> <u>modul-hc-06-podklyuchenie-k-arduino-upravlenie-ustroystvami-s-</u> telefona
- 4. https://microtechnics.ru/raspberry-pi-obmen-dannymi-po-interfejsuspi/

